AMENDMENTS TO THE CLAIMS

Please amend the claims of the present application as set forth below. In accordance with the PTO 's revised amendment format, a detailed listing of all claims has been provided. This listing of claims will replace all prior versions and listings of claims in the application. Changes to the claims are shown by strikethrough (for deleted matter) and underlining (for added matter).

By way of overview claims 1, 4-19, 21-27, 29-31, and 33-42 are currently pending. More specifically, the status of the claims is indicated below:

a) Claims 1, 4, 8, 19, 21, 26, 27, 31, and 35 are currently amended;

b) Claims 5, 7, 9-18, 22-25, 29, 30, 33, 34, and 36-39 are original;

c) Claims 6, and 40-42 are previously presented; and

d) Claims 2, 3, 20, 28, and 32 are canceled without prejudice or disclaimer.

Listing of Claims

1. (Currently amended) A system comprising:

a pluggable security policy enforcement module configured to be replaceable in the system and to provide different granularities of control for a business logic in the system, wherein the business logic processes requests submitted to the system, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain,

wherein the pluggable security policy enforcement module is configured to determine, for a particular granularity of control, whether to permit an operation, requested by a user based at least in part on a permission assigned to the user, and wherein the different granularities of control comprise a plurality of sets of rules that can be replaced with each other without altering the business logic.

2. (Cancelled).

3. (Cancelled).

4. (Currently amended) A system comprising:

a pluggable security policy enforcement module configured to be replaceable in the system and to provide different granularities of control for a business logic in the system, wherein the business logic processes requests submitted to the system, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain,

wherein the pluggable security policy enforcement module includes a control module configured to determine whether to permit an operation based at least in part on accessing the business logic to identify one or more additional tests to perform, and further configured to perform the one or more additional tests.

5. (Original) A system as recited in claim 4, wherein the control module is further configured to return a result of the determining to the business logic.

6. (Previously presented) A system comprising:

a pluggable security policy enforcement module configured to be replaceable in the system and to provide different granularities of control for a business logic in the system, wherein the business logic processes requests submitted to the system, wherein the different granularities of control comprise a plurality of sets of rules, and wherein each set of rules includes a plurality of permission assignment objects, wherein each of the permission assignment objects associates a user with a particular role, wherein each particular role is associated with one or more permissions, and wherein each of the one or more permissions identifies a particular operation and context on which the operation is to be performed.

7. (Original) A system as recited in claim 6, wherein each of the permission assignment objects further identifies whether the one or more permissions in the particular role are granted to the user or denied to the user.

8. (Currently amended) One or more computer-readable media comprising computer-executable instructions that, when executed, direct a processor to perform acts including:

receiving a request to perform an operation;

checking whether to access a business logic in order to generate a result for the requested operation, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain;

obtaining, from the business logic, a set of zero or more additional tests to be performed in order to generate the result;

performing each additional test in the set of tests if there is at least one test in the set of tests;

checking a set of pluggable rules to determine the result of the requested operation; and returning, as the result, a failure indication if checking the business logic or checking the set of pluggable rules indicates that the result is a failure, otherwise returning, as the result, a success indication.

9. (Original) One or more computer-readable media as recited in claim 8, wherein the receiving comprises receiving, from the business logic, the request to perform the operation.

10. (Original) One or more computer-readable media as recited in claim 8, wherein the receiving comprises receiving, as part of the request, an indication of a user, and wherein the checking the set of pluggable rules comprises comparing an object associated with the user to the rules in the set of pluggable rules and determining whether the operation can be performed based at least in part on whether the user is permitted to perform the operation.

11. (Original) One or more computer-readable media as recited in claim 8, wherein the receiving comprises having one of a plurality of methods invoked.

12. (Original) One or more computer-readable media as recited in claim 8, wherein the set of pluggable rules is a set of security rules defined using high-level permission concepts.

13. (Original) One or more computer-readable media as recited in claim 12, wherein the high-level permission concepts include an operation and a context, wherein

the operation allows identification of an operation to be performed and the context allows identification of what the operation is to be performed on.

14. (Original) One or more computer-readable media as recited in claim 8, wherein the computer-executable instructions are implemented as an object.

15. (Original) One or more computer-readable media as recited in claim 8, wherein the computer-executable instructions further direct the processor to perform acts including:

determining if at least one of the tests in the set of zero or more additional tests would indicate a result of failure; and

returning, as the result, the failure indication without checking the set of pluggable rules.

16. (Original) One or more computer-readable media as recited in claim 8, wherein the set of pluggable rules can be replaced with another set of pluggable rules without altering the business logic.

17. (Original) One or more computer-readable media as recited in claim 8, wherein the set of pluggable rules includes a plurality of permission assignment objects, wherein each of the permission assignment objects associates a user with a particular role, wherein each particular role is associated with one or more permissions, and wherein each of the one or more permissions identifies a particular operation and context on which the operation is to be performed.

18. (Original) One or more computer-readable media as recited in claim 17, wherein each of the permission assignment objects further identifies whether the one or more permissions in the particular role are granted to the user or denied to the user.

19. (Currently amended) A method comprising:

providing high-level permission concepts for security rules;

allowing a set of security rules to be defined using the high-level permission concepts, wherein the set of security rules allows permissions to be assigned to users of an application; and

determining, based at least in part on a permission assigned to a user, whether to permit an operation based on a request by the user,

wherein the determining further comprises determining whether to permit the operation requested by the user based at least in part on accessing a business logic to identify one or more additional tests to perform, and further comprising performing the one or more additional tests, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain.

20. (Canceled).

21. (Currently amended) A method as recited in claim [[~~20~~]] 19, further comprising returning a result of the determining to the business logic.

22. (Original) A method as recited in claim 19, wherein the high-level permission concepts include an operation and a context, wherein the operation allows identification

of an operation to be performed and the context allows identification of what the operation is to be performed on.

23. (Original) A method as recited in claim 19, wherein the method is implemented in an object having a plurality of interfaces for requesting a determination as to whether to permit a plurality of operations including the operation requested by the user.

24. (Original) A method as recited in claim 19, wherein the set of security rules includes a plurality of permission assignment objects, wherein each of the permission assignment objects associates a user with a particular role, wherein each particular role is associated with one or more permissions, and wherein each of the one or more permissions identifies a particular operation and context on which the operation is to be performed.

25. (Original) A method as recited in claim 24, wherein each of the permission assignment objects further identifies whether the one or more permissions in the particular role are granted to the user or denied to the user.

26. (Currently amended) A method comprising:

receiving a request to perform an operation associated with business logic, wherein the business logic contains problem-solving logic that produces solutions for a particular problem domain;

accessing a set of low-level rules, wherein the low-level rules are defined in terms of high-level concepts;

checking whether a user requesting to perform the operation is entitled to perform the operation based at least in part on the set of low-level rules; and

returning an indication of whether the operation is allowed or not allowed,

wherein the set of low-level rules can be replaced with another set of low-level rules without altering the business logic.

27. (Currently amended) A method as recited in claim 26, wherein the checking further comprises checking whether the user is entitled to perform the operation based at least in part on accessing [[a]] the business logic to identify one or more additional tests to perform, and further comprising performing the one or more additional tests.

28. (Canceled).

29. (Original) A method as recited in claim 27, further comprising returning the indication to the business logic.

30. (Original) A method as recited in claim 26, wherein the low-level rules include a plurality of permission assignment objects, wherein each of the permission assignment objects associates a user with a particular role, wherein each particular role is associated with one or more permissions, and wherein each of the one or more permissions identifies a particular operation and context on which the operation is to be performed

31. (Currently amended) A method comprising:

assigning high level security concepts to an application domain; and allowing a set of pluggable rules to define low-level rules, in terms of the high level security concepts, for different business logic in the application domain, wherein each business logic contains problem-solving logic that produces solutions for a particular problem domain,

wherein the high level security concepts include an operation that identifies an operation to be performed, and a context that identifies what the operation is performed on.

32. (Canceled).

33. (Original) A method as recited in claim 31, further comprising:

determining, based at least in part on a permission assigned to a user and on one or more additional tests identified by accessing the business logic, whether to permit an operation based on a request by the user

34. (Original) A method as recited in claim 33, further comprising returning a result of the determining to the business logic.

35. (Currently amended) An architecture comprising:

a plurality of resources;

a business logic layer to process, based at least in part on the plurality of resources, requests received from a client, wherein the business logic layer contains problem-solving logic that produces solutions for a particular problem domain; and a pluggable security policy enforcement module, separate from the business logic layer, to enforce security restrictions on accessing information stored at the plurality of resources.

36. (Original) An architecture as recited in claim 35, wherein the pluggable security policy enforcement module defines high-level permission concepts for security rules and further defines a set of security rules using the high-level permission concepts.

37. (Original) An architecture as recited in claim 36, wherein the high-level permission concepts include an operation and a context, wherein the operation allows identification of an operation to be performed and the context allows identification of what the operation is to be performed on.

38. (Original) An architecture as recited in claim 35, wherein the pluggable security policy enforcement module can be replaced with another pluggable security policy enforcement module to enforce different security restrictions without altering the business logic layer.

39. (Original) An architecture as recited in claim 35, wherein the pluggable security policy enforcement module is configured to determine, based at least in part on a permission assigned to a user and on one or more additional tests identified by accessing the business logic layer, whether to permit an operation to access information at the plurality of resources.

40. (Previously presented) A system as recited in claim 1, wherein the system is configured as a multi-layer architecture, wherein the business logic is implemented as a business logic layer of the multi-layer architecture.

41. (Previously presented) A system as recited in claim 1, wherein the pluggable security policy enforcement module is configured to receive an input from the business logic in the form of a user indication and an item indication.

42. (Previously presented) A system as recited in claim 1, wherein the pluggable security policy module includes an interface that provides the following interface functionality:

first functionality for testing whether an identified item can be approved by a specified user;

second functionality for testing whether the identified item of a specified type can be created by the specified user;

third functionality for testing whether the identified item can be deleted by the specified user;

fourth functionality for testing whether the identified item can be modified by the specified user; and

fifth functionality for testing whether the identified user can examine details of the identified item.